

University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

Faculty Publications from the Department of
Electrical and Computer Engineering

Electrical & Computer Engineering, Department
of

12-26-1996

Trade-off study and computer simulation for assessing spacecraft pointing accuracy and stability capabilities

Marcelo C. Algrain

Richard M. Powers

Follow this and additional works at: <https://digitalcommons.unl.edu/electricalengineeringfacpub>



Part of the [Computer Engineering Commons](#), and the [Electrical and Computer Engineering Commons](#)

This Article is brought to you for free and open access by the Electrical & Computer Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in Faculty Publications from the Department of Electrical and Computer Engineering by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

Trade-off study and computer simulation for assessing spacecraft pointing accuracy and stability capabilities

Marcelo C. Algrain, MEMBER SPIE

Richard M. Powers

University of Nebraska-Lincoln

Department of Electrical Engineering and

Center for Electro-Optics

Lincoln, Nebraska 68588-0511

E-mail: malgrain@unlinfo.unl.edu

Abstract. A case study, written in a tutorial manner, is presented where a comprehensive computer simulation is developed to determine the driving factors contributing to spacecraft pointing accuracy and stability. Models for major system components are described. Among them are spacecraft bus, attitude controller, reaction wheel assembly, star-tracker unit, inertial reference unit, and gyro drift estimators (Kalman filter). The predicted spacecraft performance is analyzed for a variety of input commands and system disturbances. The primary deterministic inputs are the desired attitude angles and rate set points. The stochastic inputs include random torque disturbances acting on the spacecraft, random gyro bias noise, gyro random walk, and star-tracker noise. These inputs are varied over a wide range to determine their effects on pointing accuracy and stability. The results are presented in the form of trade-off curves designed to facilitate the proper selection of subsystems so that overall spacecraft pointing accuracy and stability requirements are met.
© 1997 Society of Photo-Optical Instrumentation Engineers. [S0091-3286(97)02005-9]

Subject terms: pointing accuracy and stability; gyros; star-trackers; Kalman filter.

Paper 23076 received July 24, 1996; revised manuscript received Dec. 19, 1996; accepted for publication Dec. 26, 1996.

1 Introduction

Current trends in space vehicles are leading to considerably smaller platforms, which present a new set of challenges and requirements. For example, because of the smaller inertias these vehicles are likely to be more sensitive to disturbances and experience higher levels of attitude jitter. Hence, the bandwidth of the inertial reference unit may need to be enhanced if it is desired to control the high-frequency attitude jitter.¹ Furthermore, due to cost and/or power considerations, it may be desirable to replace conventional gyroscopes with quartz rate sensors or accelerometers,² and star-trackers with global positioning system (GPS) receivers configured to measure attitude. This multitude of options establishes a high priority for developing computer-aided design and analysis methods for pointing and tracking systems. The scope of the work presented in this paper is to develop a comprehensive computer simulation program capable of assessing the driving factors determining overall pointing accuracy and stability on a spacecraft (S/C). A principal objective for simulation is to provide a versatile platform that could serve as a test-bed for studying various stabilization, pointing, and tracking methods. The programming philosophy follows a modular/building-block approach where an S/C system could be quickly modeled by cutting and pasting subsystem level components from a previously developed library. This simulation is written in XMATH/SystemBuild from Integrated Systems, Inc., and the computer environment is a UNIX workstation (IBM RISC-6000). The mathematical models for the components used in the simulation are described in the sections that follow. To avoid becoming

overwhelmed by mathematical modeling, the system models used in the simulation are simplified as much as possible. However, they are quite effective in capturing the primary characteristics of each component used in the simulation, and they can be easily enhanced if the need arises.

2 Model Description

The block diagram for a generic spacecraft pointing system is shown in Fig. 1. The blocks labeled ZOH are zero-order holds. The deterministic inputs to the system are the desired S/C inertial attitude angles and rates, θ_{SP} and ω_{SP} , respectively. These desired values should be constant during pointing mode, and time varying when tracking. The primary stochastic inputs to the system are (one per axis): the random torque disturbances acting along the principal axes of inertia, the gyro random walk noises, the gyro random bias noises, and the star-tracker random bias noises. The method used to numerically integrate the system dynamics is a fourth-order Runge-Kutta³ algorithm with an integration step size of 10^{-3} s.

2.1 Spacecraft Model

The S/C bus is modeled as a rigid-body having J_x , J_y , and J_z as the polar moments of inertia along the principal axes of rotation. The angular motion of the S/C is described by Euler's equation.⁴ These relationships specify the angular accelerations along the principal axes of inertia as

$$\alpha_x = (J_y - J_z)J_x^{-1}\omega_y\omega_z + M_x J_x^{-1} \quad (\text{roll}),$$

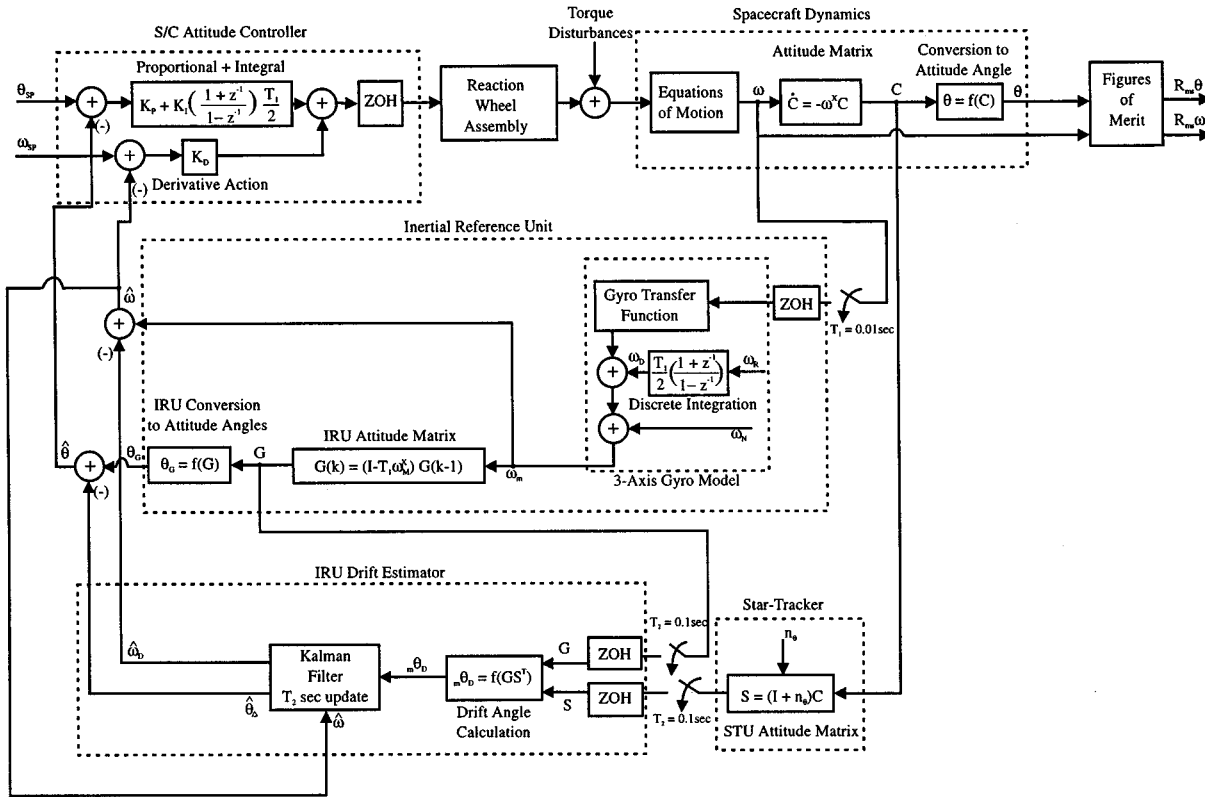


Fig. 1 Overall control system block diagram.

$$\alpha_y = (J_z - J_x)J_y^{-1}\omega_x\omega_z + M_yJ_y^{-1} \quad (\text{pitch}), \quad (1)$$

$$\alpha_z = (J_x - J_y)J_z^{-1}\omega_x\omega_y + M_zJ_z^{-1} \quad (\text{yaw}),$$

where

$$\begin{aligned}\alpha_x &= \text{angular acceleration along the roll axis} \\ \alpha_y &= \text{angular acceleration along the pitch axis} \\ \alpha_z &= \text{angular acceleration along the yaw axis} \\ \omega_x &= \text{angular velocity along the roll axis} \\ \omega_y &= \text{angular velocity along the pitch axis} \\ \omega_z &= \text{angular velocity along the yaw axis} \\ M_x &= \text{torque applied along the roll axis} \\ M_y &= \text{torque applied along the pitch axis} \\ M_z &= \text{torque applied along the yaw axis.}\end{aligned}$$

The torques acting on the S/C represent the combined effect of internal as well as external disturbances. These disturbances are modeled as zero mean random variables having Gaussian distribution. The angular velocities are computed by direct integration of the angular accelerations along the corresponding axes. Conversely, the angular displacements cannot be computed by direct integration of the angular velocities. This is a well-known fact from 3-D rotational dynamics.⁵ To obtain these angles, the S/C attitude matrix \mathbf{C} (in body coordinates) needs to be computed by solving the differential equation given below, where ω^x is

the matrix resulting from applying the cross-product operation to the angular velocity vector $\omega = [\omega_x \omega_y \omega_z]^T$.

$$\dot{\mathbf{C}} = -\boldsymbol{\omega} \times \mathbf{C} = - \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \mathbf{C}. \quad (2)$$

Once a value of the attitude matrix \mathbf{C} is obtained, the sought attitude angles can be extracted from it using a suitable Euler's angle convention.⁶ For simplicity, it is hereby assumed that we are primarily interested in measuring attitude jitter (deviations from a desired value). Also, it is assumed that the S/C frame is initially aligned with an inertial reference frame, and that the initial S/C attitude matrix (\mathbf{C} at $t=0$) is the identity matrix (positional reference). Then, small angle approximations are applicable, and the S/C attitude angles (in the roll, pitch, and yaw axes) can be extracted from the resulting attitude matrix as follows:

$$\theta \equiv f(\mathbf{C}) = [\theta_x \theta_y \theta_z]^T, \quad (3)$$

where $\theta_x = \frac{1}{2}(C_{23} - C_{32})$, $\theta_y = \frac{1}{2}(C_{31} - C_{13})$, $\theta_z = \frac{1}{2}(C_{12} - C_{21})$, where C_{ij} is an element of matrix \mathbf{C} in row i and column j , and f is defined as the operator that extracts the components of θ from the matrix \mathbf{C} .

2.2 Star-Tracker Unit

The star-tracker unit (STU) provides values of the S/C attitude matrix at the rate of T_2 (slow rate). Although an STU is typically very accurate, the small errors associated with its measurements can be modeled as additive noise/inaccuracies. These errors result from considering that the attitude matrix produced by the STU, namely \mathbf{S} , differs from the true S/C attitude \mathbf{C} by an additional rotation defined as the error term n_θ^\times . This leads to the approximation

$$\mathbf{S} = (\mathbf{I} + n_\theta^\times) \mathbf{C}, \quad (4)$$

where \mathbf{S} is the attitude matrix given by the star-tracker, \mathbf{I} is a 3×3 identity matrix, n_θ^\times denotes the cross-product operation applied to measurement error vector \mathbf{n}_θ , and \mathbf{C} is the S/C true attitude matrix. The three orthogonal components of the measurement noise vector \mathbf{n}_θ^\times are modeled as uncorrelated zero-mean Gaussian processes. Then, by specifying the $1-\sigma$ accuracy (per axis) of the star-tracker to ϵ arcsec, we mean that the standard deviation of each components in \mathbf{n}_θ is ϵ arcsec.

2.3 Inertial Reference Unit

The inertial reference unit (IRU) primarily consists of three orthogonal gyros providing S/C angular velocities measurements at the rate T_1 (fast rate). Each of the gyro outputs is modeled as a noise-corrupted low-pass-filtered version of the actual S/C angular rates. The low-pass filter is essentially the gyro transfer function. The noise sources correspond to a random drift rate and a random bias rate. This model is mathematically represented by the following Laplace transformed equation:

$$\omega_M = \mathbf{H}_{\text{gyro}} \omega + \omega_D + \omega_N, \quad (5)$$

where ω_M is the measured angular velocity vector, ω is the actual S/C angular velocity vector, ω_D is the random drift rate vector, ω_N is the random bias rate vector, and \mathbf{H}_{gyro} is the gyro transfer function. The random drift rate vector ω_D is generated by integrating the random drift noise vector ω_R . Each component of ω_R is a zero-mean random variable having Gaussian distribution. The random bias vector ω_N is also a zero-mean random variable having Gaussian distribution. Based on this model, the accuracy of the gyro measurements are determined by the standard deviation of the random processes ω_R and ω_N . The attitude matrix calculated using IRU data can be obtained using the same equation form given in Eq. (2). Then, the IRU attitude matrix \mathbf{G} (in body coordinates) can be obtained by numerically integrating the following differential equation:

$$\dot{\mathbf{G}} = -\omega_M^\times \mathbf{G}. \quad (6)$$

However, since the gyro measurements are updated at a relatively fast rate (every T_1 s), differential Eq. (6) can be approximated as the first-order difference Eq. (7) where k denotes the time index:

$$\mathbf{G}(k+1) = (\mathbf{I} - T_1 \omega_M^\times) \mathbf{G}(k). \quad (7)$$

For small angles, the IRU measured attitude angles can be extracted from the attitude matrix using the approach given in Eq. (3), as

$$\begin{aligned} \theta_{G_x} &= \frac{1}{2}(G_{23} - G_{32}), \\ \theta_{G_y} &= \frac{1}{2}(G_{31} - G_{13}), \\ \theta_{G_z} &= \frac{1}{2}(G_{12} - G_{21}). \end{aligned} \quad (8)$$

The initial IRU attitude matrix \mathbf{G} ($t=0$) is assumed to be the identity matrix. In other words, the orthogonal axes of the S/C , STU, and IRU are assumed to be initially aligned.

2.4 IRU Drift Estimator

Gyro drift introduces errors in the attitude measurements produced by an IRU. A Kalman filter can be used to estimate and neutralize the attitude errors caused by gyro drift.⁷⁻⁹ A model describing the discrete-time propagation of the drift errors is given by

$$\mathbf{X}(k+1) = \mathbf{A}(k) \mathbf{X}(k) + \mathbf{B}(k) \mathbf{U}(k) \quad (9)$$

with

$$\mathbf{X} = [\theta_{D_x} \ \theta_{D_y} \ \theta_{D_z} \ \omega_{D_x} \ \omega_{D_y} \ \omega_{D_z}]^T,$$

and θ_{D_x} is the roll drift angle, θ_{D_y} is the pitch drift angle, θ_{D_z} is the yaw drift angle, ω_{D_x} is the roll drift rate, ω_{D_y} is the pitch drift rate, ω_{D_z} is the yaw drift rate, with

$$\mathbf{U} = [\omega_{N_x} \ \omega_{N_y} \ \omega_{N_z} \ \omega_{R_x} \ \omega_{R_y} \ \omega_{R_z}]^T,$$

and ω_{N_x} is the roll drift rate random bias, ω_{N_y} is the pitch drift rate random bias, ω_{N_z} is the yaw drift rate random bias, ω_{R_x} is the roll drift random noise, ω_{R_y} is the pitch drift random noise, and ω_{R_z} is the yaw drift random noise, with

$$\mathbf{A} = \begin{bmatrix} 1 & T_2 \omega_z & -T_2 \omega_y & T_2 & 0 & 0 \\ -T_2 \omega_z & 1 & T_2 \omega_x & 0 & T_2 & 0 \\ T_2 \omega_y & -T_2 \omega_x & 1 & 0 & 0 & T_2 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{B} = T_2 \mathbf{I}_{6 \times 6},$$

and T_2 is the STU update time interval and k is the time index. The random inputs to the model are the Gaussian processes ω_R and ω_N . The state variables to be estimated are the vectors θ_D and ω_D . What is available from measurements is a noisy version of the drift angle vector θ_D . Thus, the model for this measurement is given by

$$\mathbf{Z}(k) = \mathbf{H} \mathbf{X}(k) + \mathbf{V}(k) = [\theta_{D_x} \ \theta_{D_y} \ \theta_{D_z}]^T + \mathbf{V}(k), \quad (10)$$

where the measurement matrix is

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix},$$

and the measurement noise vector is

$$\mathbf{V} = [\eta_{\theta_x} \ \eta_{\theta_y} \ \eta_{\theta_z}]^T,$$

where η_{θ_x} is the STU roll attitude angle measurement noise, η_{θ_y} is the STU pitch attitude angle measurement noise, and η_{θ_z} is the STU yaw attitude angle measurement noise. The measured drift angle ${}_m\theta_D$ is obtained indirectly. It results from considering that the IRU attitude matrix differs from the S/C attitude matrix by an additional rotation attributed to an error term defined as δ^\times . Assuming a fairly accurate STU, the S/C attitude matrix could be substituted by the STU attitude matrix. This yields the following approximation:

$$\mathbf{G} = (\mathbf{I} + \delta^\times) \mathbf{C} \cong (\mathbf{I} + \delta^\times) \mathbf{S}. \quad (11)$$

Multiplying both sides of the approximation by \mathbf{S}^{-1} , and recalling that $\mathbf{S}^{-1} = \mathbf{S}^T$, it follows that

$$\delta^\times = \mathbf{I} - \mathbf{G} \mathbf{S}^T. \quad (12)$$

Then, the measurement for the IRU attitude error is given by the equations below, where GS_{ij}^T represents the element in row i and column j of the matrix resulting from multiplying matrices \mathbf{G} and \mathbf{S}^T .

$${}_m\theta_D \cong f(-\delta^\times) = -f(\mathbf{G} \mathbf{S}^T) = [{}_m\theta_{D_x} \ {}_m\theta_{D_y} \ {}_m\theta_{D_z}]^T, \quad (13)$$

with ${}_m\theta_{G_x} = -\frac{1}{2}(GS_{23}^T - GS_{32}^T)$, ${}_m\theta_{G_y} = -\frac{1}{2}(GS_{31}^T - GS_{13}^T)$, and ${}_m\theta_{G_z} = -\frac{1}{2}(GS_{12}^T - GS_{21}^T)$.

There are several sources contributing to the IRU attitude error. Among them are the discrete time integration of the attitude matrix \mathbf{G} , the random bias rate ω_N , and the random drift rate ω_D . The first two are considered to be small contributors, because of the relatively small integration time step T_1 and ω_N being zero mean. On the other hand, the drift rate ω_D is not zero mean in the short term, and thus it is the major contributor to the IRU body attitude error. Therefore, ${}_m\theta_D$ is essentially a measurement of the drift angle. The STU accuracy also plays a role in the drift angle measurement. The error introduced by the accuracy of the STU is essentially the confidence in this measurement. Thus, the standard deviation of drift angle measurement noise is nearly equal to the $1-\sigma$ STU accuracy. This was corroborated by many independent computer simulation runs.

The Kalman filter within the IRU drift estimator provides the best available estimates of the attitude error \mathbf{X} [see Eq. (9)] based on the noisy drift angle measurement \mathbf{Z} . This filter is realized by repeated execution of the calculations given below.

Kalman gain:

$$\mathbf{K}(k) = \mathbf{P}(k|k-1) \mathbf{H}^T [\mathbf{H} \mathbf{P}(k|k-1) \mathbf{H}^T + \mathbf{R}(k)]^{-1}.$$

State estimate update:

$$\hat{\mathbf{X}}(k) = \hat{\mathbf{X}}(k|k-1) + \mathbf{K}(k) [\mathbf{Z}(k) - \mathbf{H} \hat{\mathbf{X}}(k|k-1)].$$

Error covariance update:

$$\mathbf{P}(k) = -[\mathbf{I} - \mathbf{K}(k) \mathbf{H}] \mathbf{P}(k|k-1).$$

Propagate for next time step:

$$\hat{\mathbf{X}}(k+1|k) = \mathbf{A}(k) \hat{\mathbf{X}}(k),$$

$$\mathbf{P}(k+1|k) = \mathbf{A}(k) \mathbf{P}(k) \mathbf{A}(k)^T + \mathbf{Q}(k), \quad (14)$$

increment k by one and repeat from the top with

$$\mathbf{R} = \begin{bmatrix} \sigma_{\text{STU}}^2 & 0 & 0 \\ 0 & \sigma_{\text{STU}}^2 & 0 \\ 0 & 0 & \sigma_{\text{STU}}^2 \end{bmatrix} \quad \text{and} \quad \mathbf{Q} = \begin{bmatrix} \sigma_{\omega_{N_x}}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\omega_{N_y}}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\omega_{N_z}}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\omega_{R_x}}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\omega_{R_y}}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{\omega_{R_z}}^2 \end{bmatrix},$$

where σ_{STU}^2 is the variance of the STU attitude angle measurement error, $\sigma_{\omega_{N_x}}^2$, $\sigma_{\omega_{N_y}}^2$, and $\sigma_{\omega_{N_z}}^2$ are the variances of the random drift noise terms, $\sigma_{\omega_{R_x}}^2$, $\sigma_{\omega_{R_y}}^2$, and $\sigma_{\omega_{R_z}}^2$ are the variances of the drift rate random bias terms.

2.5 S/C Attitude Controller

The 3-D attitude controller consists of three independent proportional plus integral plus derivative (PID) controllers (one per axis). Instead of generating a pseudo-rate based on angular displacements, the estimated rate was directly fed into the derivative control gain, as shown in Fig. 1. It was found that this approach was more accurate than using pseudo-rates, although the improvement was not large. The proportional, integral, and derivative gains (K_P , K_I , and K_D , respectively) are calculated as follows:

$$K_P = J \omega_C^2, \quad K_I = 0.1 J \zeta \omega_C^3, \quad K_D = 2 J \zeta \omega_C, \quad (15)$$

where J is the moment of inertia for the given axis, ζ is the damping coefficient, and ω_C is the controller bandwidth.

The discrete-time implementation of the PID controller has the update rate of T_1 s (same as for the IRU). The

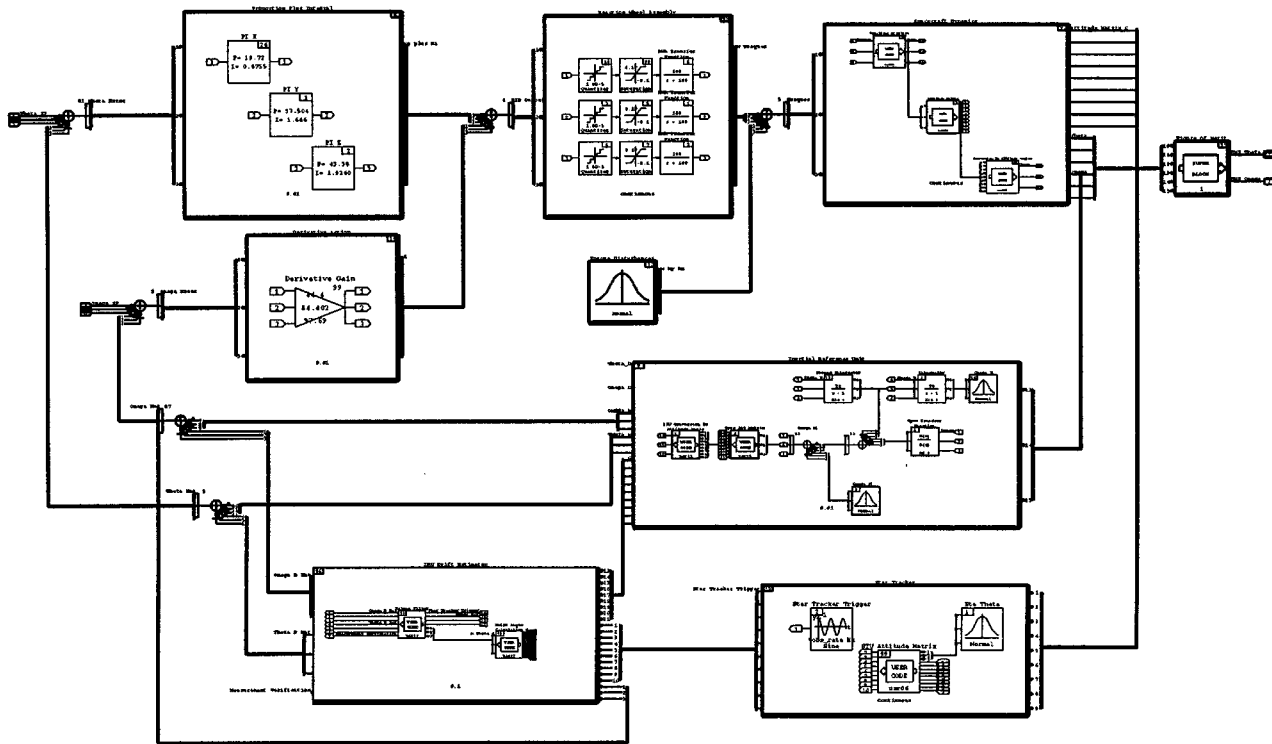


Fig. 2 XMath/SystemBuild simulation diagram.

inputs to the PID are the attitude angle and rate errors. These errors are generated by subtracting the estimated *S/C* attitude angles and rates from their respective set-points, as shown in Fig. 1. The estimated *S/C* attitude angles are calculated by subtracting the estimated drift angles (provided by the Kalman filter) from the measured IRU attitude angles (see Fig. 1). Likewise, the estimated *S/C* angular rates are obtained by subtracting the estimated gyro drift rates from the angular rates measured by the IRU (see Fig. 1). The outputs of the PID controller are then used as the command signals that regulate the operation of the reaction wheels.

2.6 Reaction Wheel Assembly

A reaction wheel assembly (RWA) typically consist of four spinning masses mounted in a tetrahedral configuration. For simplicity, and without loss of generality, the RWA is modeled here as three orthogonally mounted reaction wheels capable of introducing torques along the principal axes of inertia. Each device is modeled as the combination of a quantizer, a saturation function, and a first-order transfer function. The reaction wheel time constant is $\tau=0.01$ s, and the saturation curve has 1:1 slope, and a bound of ± 0.1 N m. The quantization step size is 10^{-5} N m.

3 Computer Simulation Program and Results

The computer simulation is implemented using the XMath/SystemBuild package from Integrated Systems, Inc. Other commercially available software programs, such as MATLAB/Simulink, could be used as well. This type software allows the user to develop a simulation using a

graphical and interactive block diagram editor for implementing design models. Also bundled with SystemBuild is a real-time block diagram simulation program that enables us to validate the models before implementation. XMath provides matrix mathematical capabilities to support the design and simulation efforts, as well as an extensive list of advanced control system development tools.

This software enables us to develop the simulation of our *S/C* pointing system by directly transferring the information on the block diagram representation given in Fig. 1. This process yields the SystemBuild diagram shown in Fig. 2. The two blocks on the upper-left corner contain the 3-D PID controller for *S/C* attitude. These blocks are followed by the three orthogonal reaction wheels forming the RWA. Directly below the RWA is the torque disturbance generator providing random inputs to the *S/C*. This is implemented using a Gaussian signal generator from the XMath library. The larger block on the upper-right corner holds the *S/C* dynamics. The *S/C* motion is described by continuous-time Euler's nonlinear equations, which are contained inside the user code block. The IRU is shown in the middle section of the simulation diagram. Within it, the gyro random drift and bias noises are generated, and the IRU attitude calculations executed. The STU block is shown in the lower-right corner of the diagram. The Kalman filter estimating IRU drift is inside the block at lower-left corner of the figure. The filter algorithm is written in C, and incorporated into the simulation via a user code block. It is clear this simulation diagram has a one-to-one correspondence with the original system block diagram given in Fig. 1. A very nice feature of constructing the

simulation using the SystemBuild graphical programming tool is that if we were to replace an STU with GPS attitude, as an example, we would only need to cut and paste the corresponding blocks. Thus, a simulation can be quickly constructed by assembling various system components extracted from an existing library.

Several computer simulation runs were made to establish subsystem requirements for a *S/C* pointing system. The pointing specifications are given in terms of accuracy and stability. For the purpose of this study, pointing accuracy is defined as the root mean square (rms) value of the error in *S/C* attitude angle. This error is defined as the difference between the desired and the actual *S/C* attitude angles. Likewise, the pointing stability is defined as the rms value of the error in *S/C* angular rates. This error is defined as the difference between the desired and the actual *S/C* angular rates. The overall pointing system requirements for this case study are 3.0 arcsec pointing accuracy, and 0.3 arcsec/s pointing stability. These requirements are representative of a fairly accurate *S/C* pointing system. To meet these requirements, the following baseline parameters were selected:

1. *S/C* principal moments of inertia:

$$\begin{aligned} J_x &= 50 \text{ kg m}^2 \text{ (roll axis),} \\ J_y &= 95 \text{ kg m}^2 \text{ (pitch axis),} \\ J_z &= 110 \text{ kg m}^2 \text{ (yaw axis).} \end{aligned}$$

2. Sampling times: $T_1 = 0.1$ s (for STU) and $T_2 = 0.01$ s (for IRU).

3. Torque disturbances M_x , M_y , and M_z have Gaussian distribution, zero mean, and standard deviation of 10^{-4} N m.

4. Gyro random bias rate ω_N has Gaussian distribution, with zero mean and standard deviation of 10^{-6} rad/s.

5. Gyro random drift noise ω_R has Gaussian distribution, zero mean and standard deviation of 10^{-6} rad/s per s.

6. The initial estimate error covariance is 10^{-5} times a 6×6 identity matrix, and the initial conditions for the Kalman filter are assumed to be zero.

7. Gyro transfer function:

$$H_{\text{gyro}}(s) = \frac{4469s + 89.22}{s^3 + 89.22s^2 + 4469s + 89.22}$$

(10-Hz low-pass filter).

8. Star-tracker accuracy of 2.0 arcsec (standard deviation of STU attitude error in measurement).

9. PID controller bandwidth of 0.1 Hz and damping factor of 0.707.

The simulation was run for 1000 s, using the parameters specified above, yielding the following expected performance:

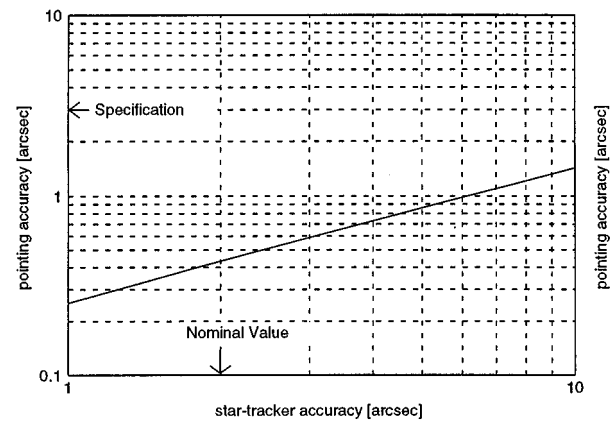


Fig. 3 Pointing accuracy versus star-tracker accuracy noise.

Pointing accuracy $\cong 0.4$ arcsec

Pointing stability $\cong 0.2$ arcsec/s.

These results satisfy the specified system requirements. To assess the influence of different parameters on pointing accuracy and stability several more runs were made. The primary parameters of interest were the star-tracker accuracy, the gyro drift noise, and the torque disturbance level. Each parameter was varied, one at a time, while keeping the others the same as for the baseline case. The first parameter to be varied was the star-tracker accuracy. The range considered was from 1 to 10 arcsec. All other parameters are as for the nominal case. The resulting pointing accuracy and stability are shown in Figs. 3 and 4, respectively. From Fig. 3, it can be concluded that the accuracy requirement is easily met, even for star-tracker accuracies as low as 10 arcsec. On the other hand, to meet the 0.3 arcsec/s stability requirement, a star-tracker with an accuracy of about 6.4 arcsec is needed (see Fig. 4).

The second parameter varied is the standard deviation of torque disturbances. The range considered is from 10^{-5} to 10^{-2} N m. From Fig. 5, it can be concluded that the accuracy requirement is easily satisfied for any value within the

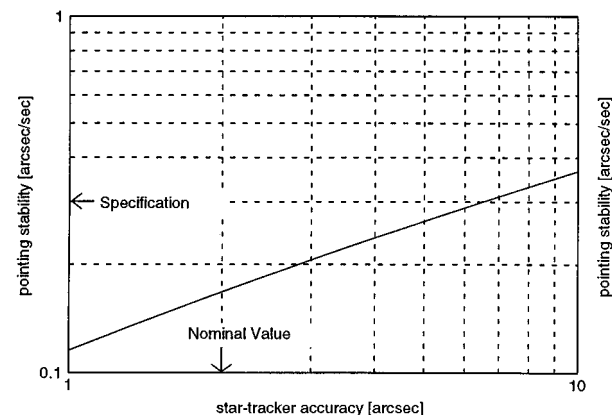


Fig. 4 Pointing stability versus star-tracker accuracy noise.

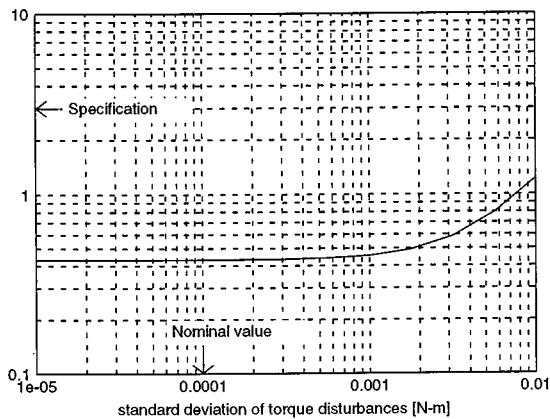


Fig. 5 Pointing accuracy versus torque disturbances.

specified range, but the pointing stability requirement restricts the value of this standard deviation of the gyro random bias noise to just over 3×10^{-3} N m (see Fig. 6).

The third parameter varied was the standard deviation of the noise source generating the gyro drift. The range under consideration is from 10^{-7} to 10^{-5} rad/s per s. From Fig. 7, it can be concluded that the accuracy requirement is satisfied for any value within the specified range, but the pointing stability requirement restricts the value of this standard deviation to less than 4×10^{-6} rad/s per s. (see Fig. 8).

The last parameter varied is the standard deviation of random bias noise. The range considered is from 10^{-7} to 10^{-4} rad/s. From Fig. 9, it can be concluded that the accuracy requirement is comfortably satisfied for any value within the specified range, but the pointing stability requirement restricts the value of this standard deviation of the gyro random bias noise to about 1.5×10^{-5} rad/s (see Fig. 10).

Additional insight into the factors affecting pointing accuracy and stability can be gained by examining the combined effects of various parameters. Because of the many possible permutations and for brevity sake, this analysis is limited to predicted pointing accuracy and stability versus

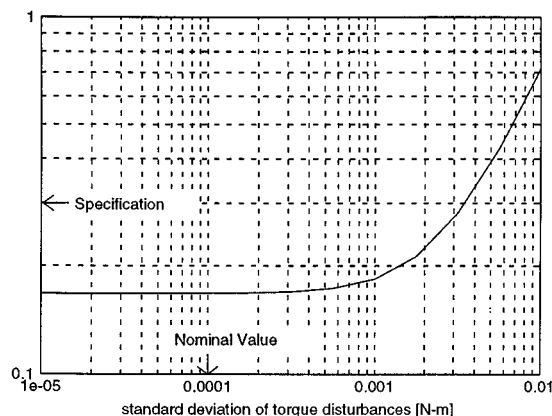


Fig. 6 Pointing stability versus torque disturbances.

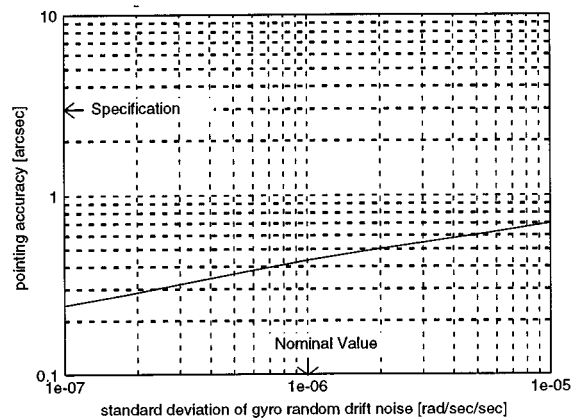


Fig. 7 Pointing accuracy versus gyro drift noise.

gyro bias and drift, pointing accuracy and stability versus STU noise and gyro drift noise, and pointing accuracy and stability versus STU noise and torque disturbance levels.

Figure 11 shows a 3-D plot for pointing accuracy when the gyro random bias and drift noises are varied simultaneously. The pointing accuracy surface shows a relatively constant slope with respect to variations on random drift noise (when plotted on a log-log scale). The effect is most noticeable at the intersection of the surface with the plane defined by the pointing accuracy and drift noise axes. Notice that taking a slice of the pointing accuracy surface along the gyro random bias value of 10^{-6} rad/s yields the curve given in Fig. 7. Turning our attention to how the gyro bias noise affects pointing stability, it is seen in Fig. 11 that the surface is relatively flat for gyro biases around 10^{-6} rad/s. For larger gyro bias levels the pointing accuracy begins to grow exponentially (when plotted in log-log scale). Slicing the pointing accuracy surface along the gyro random drift noise value of 10^{-6} rad/s per s yields the curve given in Fig. 8.

Another interesting location in the pointing accuracy surface in Fig. 11 is located at the coordinates $(10^{-5}, 10^{-5})$. The predicted accuracy is 0.6 arcsec. Notice that if the drift noise is fixed at 10^{-5} rad/s per s, no significant improvement in accuracy is obtained by reducing the

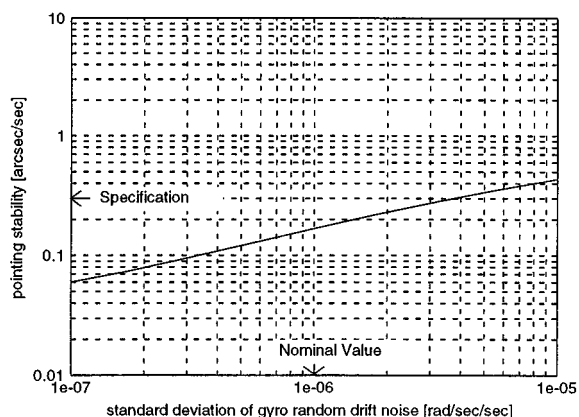


Fig. 8 Pointing stability versus gyro drift noise.

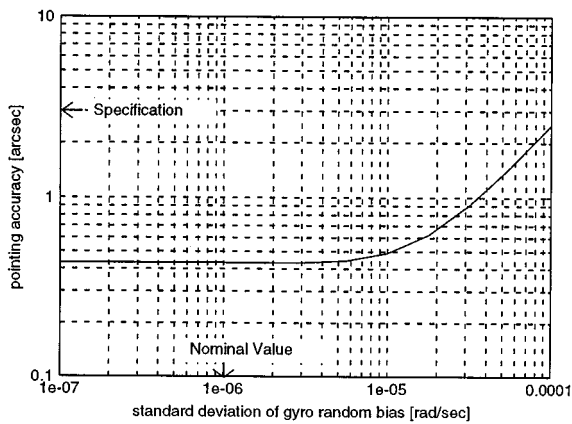


Fig. 9 Pointing accuracy versus gyro bias noise.

gyro random bias noise. However, the pointing accuracy can be improved from 0.6 to 0.4 arcsec if the drift noise is reduced from 10^{-5} to 10^{-7} rad/s per s, while keeping the gyro bias noise at the 10^{-5} rad/s level. On the other hand, if the gyro bias noise is 10^{-4} rad/s, reducing the drift noise from 10^{-5} to 10^{-7} rad/s per s yields negligible improvements in overall pointing accuracy. The results for pointing stability are given in Fig. 12. The pointing stability surface has the same type of characteristics seen for the pointing accuracy surface. For that reason, a discussion is omitted.

Figure 13 shows the 3-D plot for pointing accuracy corresponding to simultaneous variations in gyro drift noise and start-tracker accuracy. As in the results presented in Figs. 3 and 7, the slope of the pointing accuracy surface remains fairly constant with respect to variations in gyro drift noise and/or star-tracker noise (when plotted on a log-log scale). The same observation extends to the pointing stability surface shown in Fig. 14. Slicing this surface along the line corresponding to gyro drift noise of 10^{-6} rad/s per s yields the trace given in Fig. 4, while doing it along the line for star-tracker noise of 2 arcsec yields the plot given in Fig. 8. The last set of plots (Figs. 15 and 16) provide the trade-off surfaces for pointing accuracy and stability versus star-tracker noise and torque disturbances. Slicing these

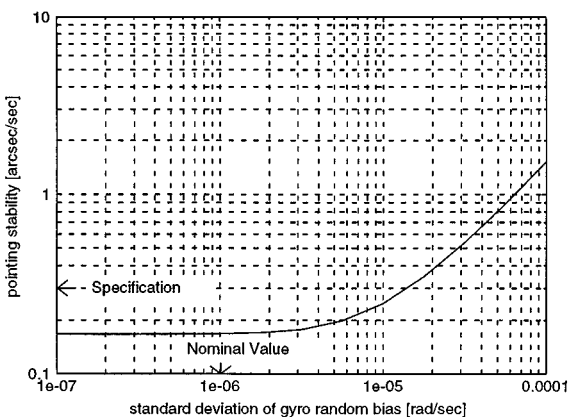


Fig. 10 Pointing stability versus gyro bias noise.

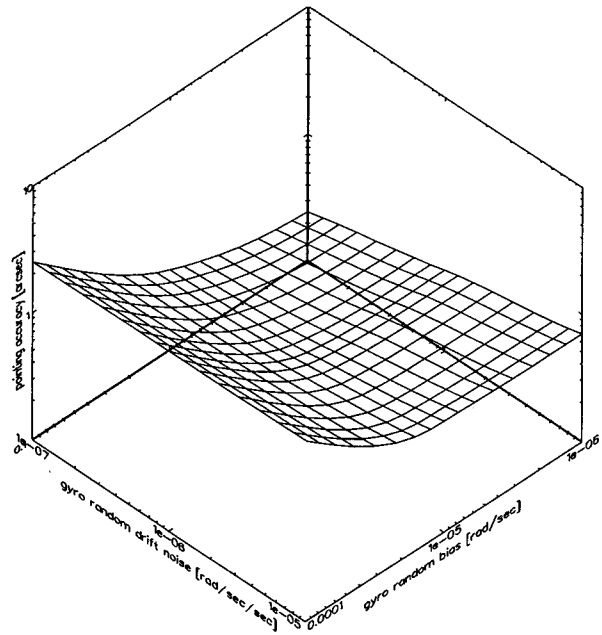


Fig. 11 Pointing accuracy for gyro bias versus gyro drift.

surfaces along the 10^{-4} N m torque disturbance line yields the plots given in Figs. 3 and 4, respectively. Likewise, slicing along the 2-arcsec STU accuracy line leads to the plots given in Figs. 5 and 6, respectively. Careful examination of the pointing accuracy surface leads to a number of interesting observations. For torque disturbances of 10^{-4} N m, reducing the STU noise from 10 to 1 arcsec improves the overall pointing accuracy from over 1 arcsec to about 0.3 arcsec, but if the torque disturbances are as high as 10^{-2} N m a more accurate STU does not apprecia-

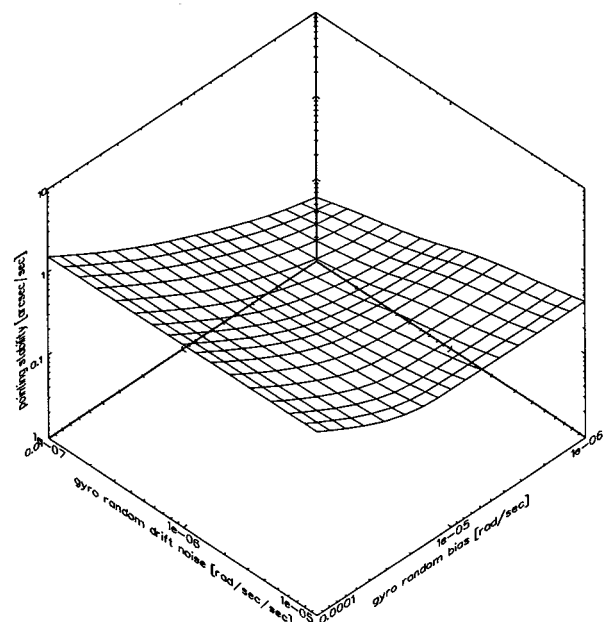


Fig. 12 Pointing stability for gyro bias versus gyro drift.

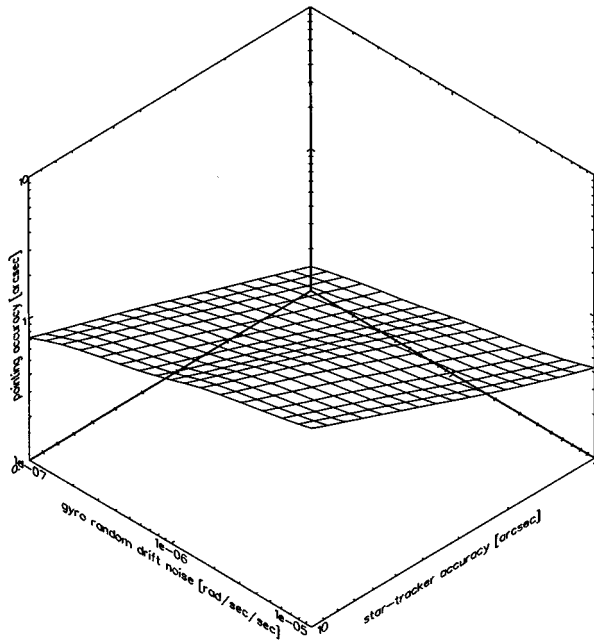


Fig. 13 Pointing accuracy for star-tracker noise versus gyro drift.

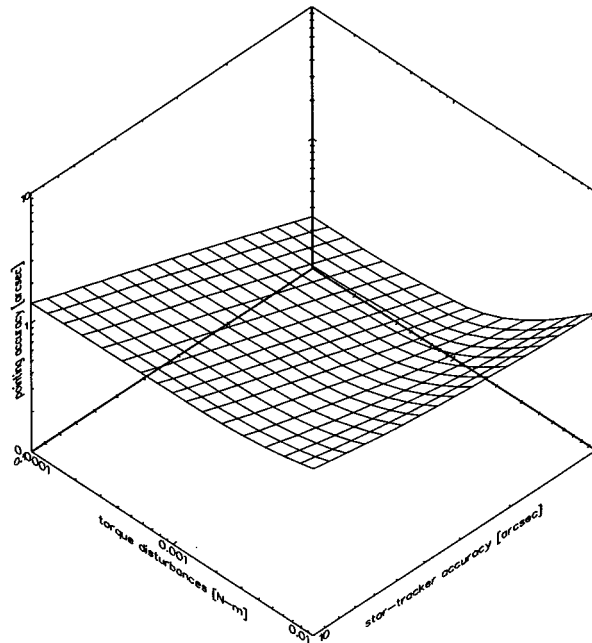


Fig. 15 Pointing accuracy for star-tracker noise versus torque disturbances.

bly improve the pointing accuracy. Also, the exponential effect seen for the pointing accuracy surface at low STU noise levels (around 1 arcsec) does not occur for higher STU noise levels (around 10 arcsec) until the torque disturbances are an order of magnitude higher. Similar conclusions can be reached from studying the pointing stability surface in Fig. 16.

4 Summary and Conclusions

The development of computer-aided design and analysis tools for precision pointing systems is presented. The central part of this work is a comprehensive computer simulation used to identify critical system components, specify

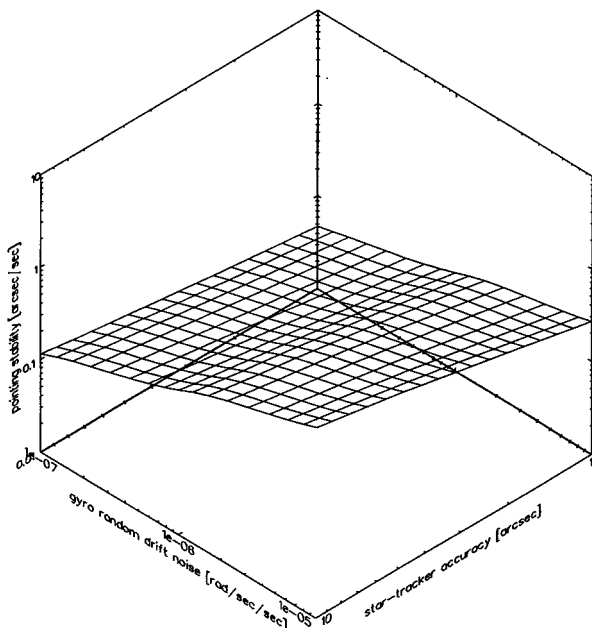


Fig. 14 Pointing stability for star-tracker noise versus gyro drift.

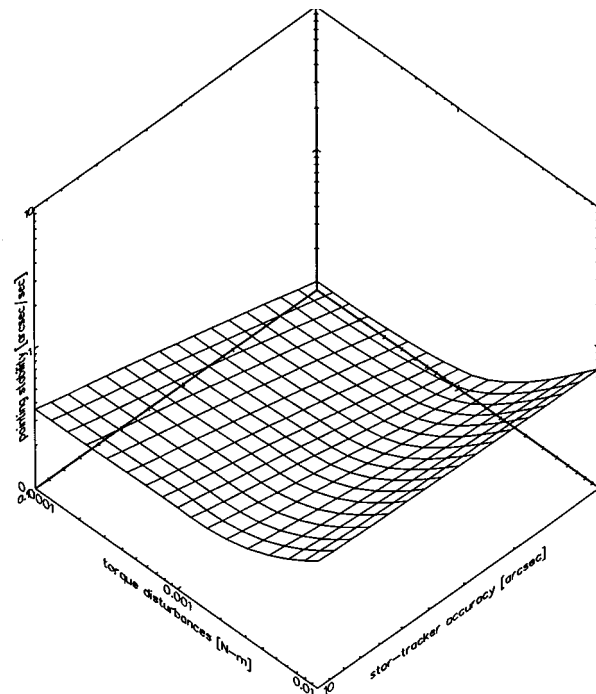


Fig. 16 Pointing stability for star-tracker noise versus torque disturbances.

subsystem requirements, and predict overall system performance. The simulation is generated using a modular/building-block approach. It provides a flexible platform where a variety of pointing system concepts can be quickly modeled. The programming language is XMATH/SystemBuild from Integrated Systems, Inc. The simulation is used in a case study to predict expected system performance of a precision spaceborne pointing system. The scope of the case study is to assess the pointing capabilities of a spacecraft, determine the expected system performance, and to conduct trade-off analyses. The nominal requirements for pointing accuracy and stability are selected as 3.0 arcsec and 0.3 arcsec/s, respectively. These values are representative of a fairly accurate S/C pointing system. The simulation results reveal that the most difficult requirement to meet is the specified pointing stability. If this requirement is met, it followed that the pointing accuracy requirement was satisfied with a generous safety margin. To establish a reference point, a baseline-line system is selected using a 2-arcsec star-tracked accuracy, 10^{-6} arcsec/s per s gyro drift acceleration, 10^{-6} arcsec/s gyro bias rate, and torque disturbances of 10^{-4} N m. For this system, the expected performance is 0.5 arcsec in pointing accuracy, and 0.2 arcsec/s in pointing stability. The system parameters are varied over a wide range of values to determine their influence on overall system performance. These effects are plotted in Figs. 3 to 10. The effects of varying two parameters at a time is demonstrated using 3-D plots for pointing accuracy and stability. The resulting surfaces are given in Figs. 11 through 16. These plots provide the basis for conducting trade-off designs to facilitate the proper and cost-effective selection of subsystem components so that overall spacecraft pointing system accuracy and stability requirements are met.

Acknowledgments

This work was supported in part by National Aeronautics and Space Administration Jet Propulsion Laboratory (JPL) through an American Society for Engineering Education (ASEE) summer faculty fellowship. We are particularly thankful to Mr. William Brekenridge, Dr. Tooraj Kia, and Dr. George Sevaston (all from JPL).

References

1. M. Algrain and M. Woehrer, "Determination of attitude jitter in small satellites," *Proc. SPIE* **2739**, 215–228 (Apr. 1996).

2. M. Algrain, "Gyroless line-of-sight stabilization for pointing and tracking systems," *Opt. Eng.* **33**(4), 1255–1260 (1994).
3. W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes in C*, Cambridge University Press, Cambridge, MA (1989).
4. F. Beer and E. Johnston, Chap. 15, "Kinematics of Rigid Bodies," in *Vector Mechanics for Engineers*, 4th ed., p. 638, McGraw-Hill, New York (1984).
5. P. Hughes, Chap. 2, "Rotational Kinematics," in *Space Attitude Dynamics*, pp. 9–29, J. Wiley & Sons, New York (1986).
6. P. Fortescue and J. Stark, *Spacecraft Systems Engineering*, John Wiley and Sons, New York (1992).
7. R. Brown and P. Hwang, Chap. 6, "Applications and Additional Topics on Discrete Kalman Filtering," in *Introduction to Random Signals and Applied Kalman Filtering*, pp. 253–257, J. Wiley & Sons, New York (1992).
8. J. Murrell, "Precision attitude determination for multimission spacecraft," in *Proc. AIAA Guidance and Control Conf.* (July 1978).
9. E. Lefferts, F. Markley, and M. Shuster, "Kalman filtering for spacecraft attitude determination," *AIAA J. Guid.* **5**, 417–429 (Sep.–Oct. 1982).



Marcelo C. Algrain received a BSME in 1980 and an MSEE in 1981, both from Wright State University, Dayton, Ohio, and in 1991 a PhD in electrical engineering from the Illinois Institute of Technology, Chicago. Since 1991 he has been an assistant professor at the University of Nebraska-Lincoln (UNL). From 1989 to 1991, he was an engineering specialist at Recon/Optical, Inc., where he performed analyses on stabilization and control systems for high-resolution reconnaissance systems. From 1984 to 1989, he was with Borg-Warner Research, developing control system prototypes for a variety of automotive and industrial products. Prior to his time with Borg-Warner, he was with Babcock & Wilcox, designing process control systems for nuclear power plants. His research interests are stabilization, pointing and tracking systems, and stochastic, optimal, and adaptive controls. Dr. Algrain has published numerous papers in the control systems field and holds two patents. He is a member SPIE, IEEE, and AIAA.



Richard M. Powers received his BSAE and MSAE degrees from the University of Colorado at Boulder in 1988 and 1991, respectively, and his MSEE degree from the University of Nebraska at Lincoln (UNL) in 1996. He is currently pursuing a PhD in electrical engineering from UNL. His technical interests include orbit determination, satellite control systems, and he is currently engaged in research on satellite instrument pointing knowledge determination.